

RRRRRRRRRRRR		UUU	UUU	NNN	NNN	000000000	FFFFFFFFFFF	FFFFFFFFFFF
RRRRRRRRRRRR		UUU	UUU	NNN	NNN	000000000	FFFFFFFFFFF	FFFFFFFFFFF
RRRRRRRRRRRR		UUU	UUU	NNN	NNN	000000000	FFFFFFFFFFF	FFFFFFFFFFF
RRR	RRR	UUU	UUU	NNN	NNN	000	FFF	FFF
RRR	RRR	UUU	UUU	NNN	NNN	000	FFF	FFF
RRR	RRR	UUU	UUU	NNN	NNN	000	FFF	FFF
RRR	RRR	UUU	UUU	NNNNNN	NNN	000	FFF	FFF
RRR	RRR	UUU	UUU	NNNNNN	NNN	000	FFF	FFF
RRR	RRR	UUU	UUU	NNNNNN	NNN	000	FFF	FFF
RRRRRRRRRRRR		UUU	UUU	NNN	NNN	000	FFFFFFFFF	FFFFFFFFF
RRRRRRRRRRRR		UUU	UUU	NNN	NNN	000	FFFFFFFFF	FFFFFFFFF
RRRRRRRRRRRR		UUU	UUU	NNN	NNN	000	FFFFFFFFF	FFFFFFFFF
RRR	RRR	UUU	UUU	NNN	NNNNNN	000	FFF	FFF
RRR	RRR	UUU	UUU	NNN	NNNNNN	000	FFF	FFF
RRR	RRR	UUU	UUU	NNN	NNNNNN	000	FFF	FFF
RRR	RRR	UUU	UUU	NNN	NNN	000	FFF	FFF
RRR	RRR	UUU	UUU	NNN	NNN	000	FFF	FFF
RRR	RRR	UUU	UUU	NNN	NNN	000	FFF	FFF
RRR	RRR	UUU	UUU	NNN	NNN	000	FFF	FFF
RRR	RRR	UUUUUUUUUUUUUUUU	NNN	NNN	000000000	FFF	FFF	
RRR	RRR	UUUUUUUUUUUUUUUU	NNN	NNN	000000000	FFF	FFF	
RRR	RRR	UUUUUUUUUUUUUUUU	NNN	NNN	000000000	FFF	FFF	

GET
V04

.....

```

LL          IIIIII          SSSSSSSS
LL          IIIIII          SSSSSSSS
LL          II             SS
LL          II             SS
LL          II             SS
LL          II             SS
LL          II             SSSSSS
LL          II             SSSSSS
LL          II             SS
LL          II             SS
LL          II             SS
LL          II             SS
LLLLLLLLLLLL IIIIII          SSSSSSSS
LLLLLLLLLLLL IIIIII          SSSSSSSS

```



```
1 0001 0 %TITLE 'Get one character'
2 0002 0 MODULE getone ( IDENT = 'V04-000'
3 P 0003 0 %BLISS32[, ADDRESSING_MODE (EXTERNAL = LONG_RELATIVE,
4 0004 0 NONEXTERNAL = LONG_RELATIVE)]
5 0005 0 ) =
6 0006 1 BEGIN
7 0007 1
8 0008 1 *****
9 0009 1 *
10 0010 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
11 0011 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
12 0012 1 * ALL RIGHTS RESERVED.
13 0013 1 *
14 0014 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
15 0015 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
16 0016 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
17 0017 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
18 0018 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
19 0019 1 * TRANSFERRED.
20 0020 1 *
21 0021 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
22 0022 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
23 0023 1 * CORPORATION.
24 0024 1 *
25 0025 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
26 0026 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
27 0027 1 *
28 0028 1 *
29 0029 1 *****
30 0030 1
31 0031 1 ++
32 0032 1 FACILITY: DSR (Digital Standard RUNOFF) / DSRPLUS
33 0033 1
34 0034 1 ABSTRACT: Get one character, applying flag recognition.
35 0035 1
36 0036 1 ENVIRONMENT: Transportable
37 0037 1
38 0038 1 AUTHOR: R.W.Friday CREATION DATE: May, 1978
39 0039 1
40 0040 1
```

GETONE
V04-000

Get one character
Revision History

G 4
16-Sep-1984 00:40:20
14-Sep-1984 13:06:33

VAX-11 Bliss-32 V4.0-742
[RUNOFF.SRC]GETONE.BLI;1

Page 2
(2)

:	42	0041	1	%SBTTL 'Revision History'
:	43	0042	1	
:	44	0043	1	MODIFIED BY:
:	45	0044	1	
:	46	0045	1	011 REM00011 Ray Marshall 17-November-1983
:	47	0046	1	Modified the external definition of ATABLE to use the new
:	48	0047	1	macro ATABLE_DEFINITION defined in ATCODE.REQ.
:	49	0048	1	
:	50	0049	1	010 RER00010 Ron Randall 07-Mar-1983
:	51	0050	1	Global edit of all modules. Updated module names, idents,
:	52	0051	1	copyright dates. Changed require files to BLISS library.
:	53	0052	1	
:	54	0053	1	--
:	55	0054	1	


```

: 57      0055 1 %SBTTL 'Module Level Declarations'
: 58      0056 1
: 59      0057 1
: 60      0058 1 | TABLE OF CONTENTS:
: 61      0059 1
: 62      0060 1 | INCLUDE FILES:
: 63      0061 1
: 64      0062 1 LIBRARY 'NXPORT:XPORT';      | XPORT Library
: 65      0063 1 REQUIRE 'REQ:RNODEF';      | RUNOFF variant definitions
: 66      0194 1
: 67      U 0195 1 %IF DSRPLUS %THEN
: 68      U 0196 1 LIBRARY 'REQ:DPLLIB';      | DSRPLUS BLISS Library
: 69      0197 1 %ELSE
: 70      0198 1 LIBRARY 'REQ:DSRLIB';      | DSR BLISS Library
: 71      0199 1 %FI
: 72      0200 1
: 73      0201 1 |
: 74      0202 1 | EXTERNAL REFERENCES:
: 75      0203 1
: 76      0204 1 EXTERNAL LITERAL
: 77      0205 1 RINTES : UNSIGNED (8);
: 78      0206 1
: 79      0207 1 EXTERNAL
: 80      0208 1 atable : atable_definition, | Action table. Used to identify what type of
: 81      0209 1 | action is to be taken on encountering any
: 82      0210 1 | given character.
: 83      0211 1 FLGT : FLAG_TABLE [FLAG_COUNT],
: 84      0212 1 IRA : FIXED_STRING,
: 85      0213 1 KHAR,
: 86      0214 1 SCA : SCA_DEFINITION;
: 87      0215 1
: 88      0216 1 EXTERNAL ROUTINE
: 89      0217 1 FNDFLG,
: 90      0218 1 RSKIPS;
: 91      0219 1
```

```
93 0220 1 GLOBAL ROUTINE GETONE (ACCEPTABLE,ALTERNATE) =
94 0221 1
95 0222 1 ++
96 0223 1 FUNCTIONAL DESCRIPTION:
97 0224 1
98 0225 1     Skip spaces and tabs until a different character is found.
99 0226 1     If that character is <CONTROL flag>, or RINTES (i.e., end of record),
100 0227 1     or a semicolon, return a zero (null character). If the character
101 0228 1     is the accept (quote) flag, apply the flag against the next
102 0229 1     character. If the character is the flag already in effect for this
103 0230 1     flag operation or a special cased ALTERNATE, take it without complaining.
104 0231 1     If the character is any other flag, then return the negative of the
105 0232 1     character's value so that the caller (OFT) may complain with RNFFAU
106 0233 1     (flag already in use).
107 0234 1
108 0235 1 FORMAL PARAMETERS:
109 0236 1
110 0237 1     ACCEPTABLE is the value of the flag character for the current
111 0238 1     .FLAGS or .NO FLAGS command.
112 0239 1
113 0240 1     ALTERNATE is the value of the flag character for an alternately
114 0241 1     acceptable flag.
115 0242 1
116 0243 1 IMPLICIT INPUTS:      None
117 0244 1
118 0245 1 IMPLICIT OUTPUTS:     None
119 0246 1
120 0247 1 ROUTINE VALUE:
121 0248 1 COMPLETION CODES:
122 0249 1
123 0250 1     The routine returns the value of the character supplied; or zero if
124 0251 1     the character was not a flag; or -1*<character-value> if the character
125 0252 1     was the same character as that used for another flag. The negative
126 0253 1     return value represents an error condition that the caller (OFT) has
127 0254 1     the responsibility of reporting.
128 0255 1
129 0256 1 SIDE EFFECTS:          None
130 0257 1 --
131 0258 1
132 0259 2 BEGIN
133 0260 2 RSKIPS (IRA);                                !Skip spaces and tabs.
134 0261 2
135 0262 2 IF .KHAR EQL RINTES
136 0263 2 THEN
137 0264 2     RETURN 0;                                !End of record.
138 0265 2
139 0266 2 IF .KHAR EQL .FLGT [CON_FLAG, FLAG_CHARACTER]
140 0267 2 THEN
141 0268 2     !<CONTROL flag>
142 0269 2     RETURN 0;                                !Start of a new command.
143 0270 2
144 0271 2 IF .ATABLE [.KHAR] EQL A_FLAG
145 0272 2 THEN
146 0273 2     !Character is a flag. Do it.
147 0274 2     BEGIN
148 0275 2     LOCAL
149 0276 2     WHICH_FLAG;
```



```
150 0277 3
151 0278 3
152 0279 3
153 0280 3
154 0281 3
155 0282 3
156 0283 3
157 0284 4
158 0285 4
159 0286 4
160 0287 4
161 0288 4
162 0289 4
163 0290 4
164 0291 4
165 0292 4
166 0293 4
167 0294 4
168 0295 4
169 0296 4
170 0297 4
171 0298 4
172 0299 4
173 0300 4
174 0301 4
175 0302 4
176 0303 4
177 0304 4
178 0305 4
179 0306 5
180 0307 5
181 0308 5
182 0309 5
183 0310 5
184 0311 5
185 0312 5
186 0313 4
187 0314 4
188 0315 3
189 0316 3
190 0317 3
191 0318 4
192 0319 4
193 0320 4
194 0321 4
195 0322 4
196 0323 4
197 0324 4
198 0325 3
199 0326 3
200 0327 3
201 0328 4
202 0329 4
203 0330 4
204 0331 4
205 0332 4
206 0333 4

WHICH_FLAG = FNDFLG (.KHAR);          !Identify the flag.

CASE WHICH_FLAG FROM 0 TO FLAG_COUNT + 1 OF
  SET
    [QUO_FLAG] :
      BEGIN
        !<QUOTE flag>
        LOCAL
          HOLD_QUO;
          HOLD_QUO = .KHAR;
          KCNS ();                      !Get next character
        IF .KHAR EQL RINTES
          THEN
            IF .WHICH_FLAG EQL .ACCEPTABLE OR
              .WHICH_FLAG EQL .ALTERNATE
              ! User is redefining <QUOTE flag> to itself or an
              ! acceptable alternate (if there is such a thing). Let him.
            THEN
              RETURN .HOLD_QUO
            ! User is trying to define some other flag to the
            ! <QUOTE flag>. Let OFT complain.
            ELSE
              RETURN -.HOLD_QUO
            ELSE
              BEGIN
                LOCAL
                  HOLD_KHAR;
                HOLD_KHAR = .KHAR;
                KCNS ();
                RETURN .HOLD_KHAR;
                END;
            END;
          [FLAG_COUNT + 1] :
            BEGIN
              LOCAL
                HOLD_CHAR;
              HOLD_CHAR = .KHAR;
              KCNS ();
              RETURN .HOLD_CHAR;
              END;
            [INRANGE] :
              BEGIN
                LOCAL
                  HOLD_CHAR;
                HOLD_CHAR = .KHAR;
                KCNST);
```

```

207 0334 4
208 0335 4 IF .WHICH_FLAG EQL .ACCEPTABLE OR
209 0336 4 .WHICH_FLAG EQL .ALTERNATE
210 0337 4 THEN
211 0338 4 !Let user redefine the flag to itself (or to the alternate).
212 0339 4 RETURN .HOLD_CHAR
213 0340 4 ELSE
214 0341 4 !Complain if he redefines it to the value of another flag.
215 0342 4 RETURN (-.HOLD_CHAR);
216 0343 4
217 0344 4 END;
218 0345 4
219 0346 4 TES;
220 0347 4
221 0348 4 END;
222 0349 4
223 0350 4 IF .KHAR EQL %C';'
224 0351 4 THEN
225 0352 4 RETURN 0;
226 0353 4
227 0354 4 BEGIN
228 0355 4 LOCAL
229 0356 4 HOLD_KHAR;
230 0357 4
231 0358 4 HOLD_KHAR = .KHAR;
232 0359 4 KCNS();
233 0360 4 RETURN .HOLD_KHAR
234 0361 4 END
235 0362 4 END;

```

```
.TITLE  GETONE Get one character
.IDENT  \V04-000\
```

.EXTRN RINTES, ATABLE, FLGT
.EXTRN IRA, KHAR, SCA, FNDFLG
.EXTRN RSKIPS

```
.PSECT $CODE$,NOWRT,2
```

```

.ENTRY GETONE, Save R2,R3,R4,R5
MOVZBL #RINTES, R5

```

```
MOVAB    KHR, R4
MOVAB    IRA+12, R3
```

PUSHAB IRA
CALLS #1, RSKIPS

```

MOVLE  KHAR, R0
CMPL   R0, R5

```

```

CMPL      R0, R0
BEQL      1$
CMPL      R0, FLGT+80

```

BNEQ	2\$
BRW	17\$
CHRD	17\$

CMPB
BEQL

BRW 15\$
PUSHL RO
CALLS #1, FNDFLG

Address	Hex Data	ASCII Data
00000000G	55	003C 00000
00000000G	54	00G 8F 9A 00002
00000000G	53	000000000G EF 9E 00006
00000000G	53	000000000G EF 9E 0000D
00000000G		F4 A3 9F 00014
00000000G	EF	01 FB 00017
00000000G	50	64 D0 0001E
00000000G	55	50 D1 00021
00000000G		07 13 00024
00000000G	EF	50 D1 00026
00000000G		03 12 0002D 1\$:
00000000G		00C5 31 0002F
00000000G	03	000000000GEF40 91 00032 2\$:
00000000G		03 13 0003A
00000000G		009B 31 0003C
00000000G		50 DD 0003F 3\$:
00000000G	EF	01 FB 00041

0220
0260
0262
0266
0271
0278

PC	Op	Op2	Op3	Op4	Op5	Op6	Op7	Op8	Op9	Op10	Op11	Op12	Op13	Op14	Op15	Op16	Op17	Op18	Op19	Op20	Op21	Op22	Op23	Op24	Op25	Op26	Op27	Op28	Op29	Op30	Op31	Op32	Op33	Op34	Op35	Op36	Op37	Op38	Op39	Op40	Op41	Op42	Op43	Op44	Op45	Op46	Op47	Op48	Op49	Op50	Op51	Op52	Op53	Op54	Op55	Op56	Op57	Op58	Op59	Op60	Op61	Op62	Op63	Op64	Op65	Op66	Op67	Op68	Op69	Op70	Op71	Op72	Op73	Op74	Op75	Op76	Op77	Op78	Op79	Op80	Op81	Op82	Op83	Op84	Op85	Op86	Op87	Op88	Op89	Op90	Op91	Op92	Op93	Op94	Op95	Op96	Op97	Op98	Op99	Op100	Op101	Op102	Op103	Op104	Op105	Op106	Op107	Op108	Op109	Op110	Op111	Op112	Op113	Op114	Op115	Op116	Op117	Op118	Op119	Op120	Op121	Op122	Op123	Op124	Op125	Op126	Op127	Op128	Op129	Op130	Op131	Op132	Op133	Op134	Op135	Op136	Op137	Op138	Op139	Op140	Op141	Op142	Op143	Op144	Op145	Op146	Op147	Op148	Op149	Op150	Op151	Op152	Op153	Op154	Op155	Op156	Op157	Op158	Op159	Op160	Op161	Op162	Op163	Op164	Op165	Op166	Op167	Op168	Op169	Op170	Op171	Op172	Op173	Op174	Op175	Op176	Op177	Op178	Op179	Op180	Op181	Op182	Op183	Op184	Op185	Op186	Op187	Op188	Op189	Op190	Op191	Op192	Op193	Op194	Op195	Op196	Op197	Op198	Op199	Op200	Op201	Op202	Op203	Op204	Op205	Op206	Op207	Op208	Op209	Op210	Op211	Op212	Op213	Op214	Op215	Op216	Op217	Op218	Op219	Op220	Op221	Op222	Op223	Op224	Op225	Op226	Op227	Op228	Op229	Op230	Op231	Op232	Op233	Op234	Op235	Op236	Op237	Op238	Op239	Op240	Op241	Op242	Op243	Op244	Op245	Op246	Op247	Op248	Op249	Op250	Op251	Op252	Op253	Op254	Op255	Op256	Op257	Op258	Op259	Op260	Op261	Op262	Op263	Op264	Op265	Op266	Op267	Op268	Op269	Op270	Op271	Op272	Op273	Op274	Op275	Op276	Op277	Op278	Op279	Op280	Op281	Op282	Op283	Op284	Op285	Op286	Op287	Op288	Op289	Op290	Op291	Op292	Op293	Op294	Op295	Op296	Op297	Op298	Op299	Op300	Op301	Op302	Op303	Op304	Op305	Op306	Op307	Op308	Op309	Op310	Op311	Op312	Op313	Op314	Op315	Op316	Op317	Op318	Op319	Op320	Op321	Op322	Op323	Op324	Op325	Op326	Op327	Op328	Op329	Op330	Op331	Op332	Op333	Op334	Op335	Op336	Op337	Op338	Op339	Op340	Op341	Op342	Op343	Op344	Op345	Op346	Op347	Op348	Op349	Op350	Op351	Op352	Op353	Op354	Op355	Op356	Op357	Op358	Op359	Op360	Op361	Op362	Op363	Op364	Op365	Op366	Op367	Op368	Op369	Op370	Op371	Op372	Op373	Op374	Op375	Op376	Op377	Op378	Op379	Op380	Op381	Op382	Op383	Op384	Op385	Op386	Op387	Op388	Op389	Op390	Op391	Op392	Op393	Op394	Op395	Op396	Op397	Op398	Op399	Op400	Op401	Op402	Op403	Op404	Op405	Op406	Op407	Op408	Op409	Op410	Op411	Op412	Op413	Op414	Op415	Op416	Op417	Op418	Op419
----	----	-----	-----	-----	-----	-----	-----	-----	-----	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

51	51	CE	000D3	MNEGL	HOLD CHAR, R1	0342
50	51	D0	000D6	14\$:	MOVL R1, R0	0328
		04	000D9	RET		
3B	64	D1	000DA	15\$:	CMPL KHAR, #59	0350
	18	13	000DD	BEQL	17\$	
50	64	D0	000DF	MOVL	KHAR, HOLD_KHAR	0358
	63	D5	000E2	TSTL	IRA+12	0359
	07	14	000E4	BGTR	16\$	
64	55	9A	000E6	MOVZBL	R5, KHAR	
63	01	CE	000E9	MNEGL	#1, IRA+12	
		04	000EC	RET		
64	F8	B3	9A	000ED	16\$:	
	F8	A3	D6	000F1	INCL IRA+4	
		63	D7	000F4	DECL IRA+12	
			04	000F6	RET	
		50	D4	000F7	17\$:	0360
			04	000F9	CLRL R0	0362
				RET		

; Routine Size: 250 bytes,

Routine Base: \$CODE\$ + 0000

; 236

; 237

; 238

0363 1

0364 1 END

0365 0 ELUDOM

!End of module

PSECT SUMMARY

Name	Bytes	Attributes
\$CODE\$	250	NOVEC,NOWRT, RD , EXE,NOSHR, LCL, REL, CON,NOPIC,ALIGN(2)

Library Statistics

File	Total	Symbols Loaded	Percent	Pages Mapped	Processing Time
\$255\$DUA28:[SYSLIB]XPORT.L32;1	590	0	0	252	00:00.1
\$255\$DUA28:[RUNOFF.SRC]DSRLIB.L32;1	1248	17	1	86	00:00.2

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LISS\$:GETONE/OBJ=OBJ\$:GETONE MSRC\$:GETONE/UPDATE=(ENH\$:GETONE)

GETONE Get one character
V04-000 Module Level Declarations

N 4
16-Sep-1984 00:40:20

VAX-11 Bliss-32 V4.0-742

Page 9

: Size: 250 code + 0 data bytes
: Run Time: 00:06.5
: Elapsed Time: 00:18.6
: Lines/CPU Min: 3390
: Lexemes/CPU-Min: 16291
: Memory Used: 86 pages
: Compilation Complete

GETC
V04-

0342

AH-BT13A-SE
 VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY